**Jet Propulsion Laboratory**
California Institute of Technology

A Hybrid Traveling Salesman Problem - Squeaky Wheel Optimization Planner for Earth Observational Scheduling

Garrett Lewellen, Christopher Davies, Amos Byon, Russell Knight, Elly Shao, Daniel Tran, and Michael Trowbridge
Jet Propulsion Laboratory, California Institute of Technology
June 16, 2017

Introduction

- What makes space-based observation scheduling hard?
    - Oversubscription: too many science requests, too few observers
    - Flexibility: too many opportunities to observe
    - Time-varying slew costs (Pralet and Verfaillie 2014; Lemaître et al. 2002)
- Problem statement
    - Find the largest value tour within a graph that has asymmetric, bidirectional edges, time-varying edge weights, cycles and revisits
- Similar problems are generally NP-hard or NP-complete (Karger, Motwani, and Ramkumar 1997; Lemaître et al. 2002; Ichoua, Gendreau, and Potvin 2003; Pinedo 2012; Hall and Magazine 1994)
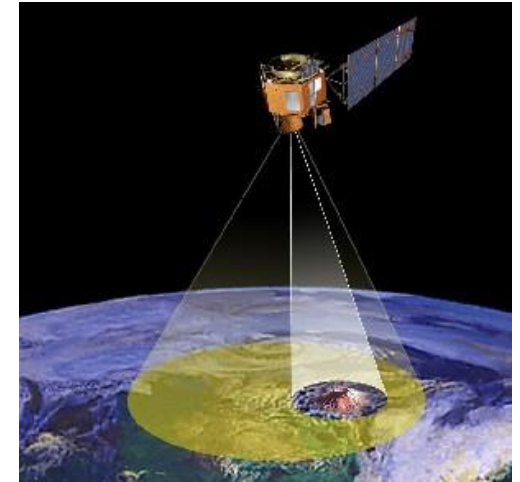
EO-1 satellite

Related work

- Squeaky Wheel Optimization (Joslin and Clements, 1999)
- Genetic algorithms: Earth Observing Satellite Scheduling Problem (Globus et al. 2004)
- Greedy stochastic search with resource-aware heuristics for the EOS Scheduling problem (Frank et al. 2001)
- Stitched window planning (Aldinger et al. 2013)
- Parallel tabu search for traffic-aware fleet vehicle routing (Ichoa, Gendreau and Potvin 2003)
- Time-dependent Simple Temporal Networks (Pralet and Verfaillie 2014)

Outline

- Formulation
- Experiments
- Results
- Discussion
- Future work
- Conclusion

jpl.nasa.gov

Formulation

Why a Hybrid Algorithm?

	Squeaky Wheel	Insertion Search (TSP)
Value optimization	Strength	Weakness
Complexity (including time, resource propagation)	$s^2+(N-s)$, $s \leq N$ Strength	N^3 Weakness
Utility, Efficiency	Weakness	Strength

- The two are complementary – combine.
- Problem: insertion search is N^3
  - Compromise: quality for speed.
  - Constraint insertion search to <N sliding window.
 - Maintain contracts at window edges as Aldinger et al. do (2013).
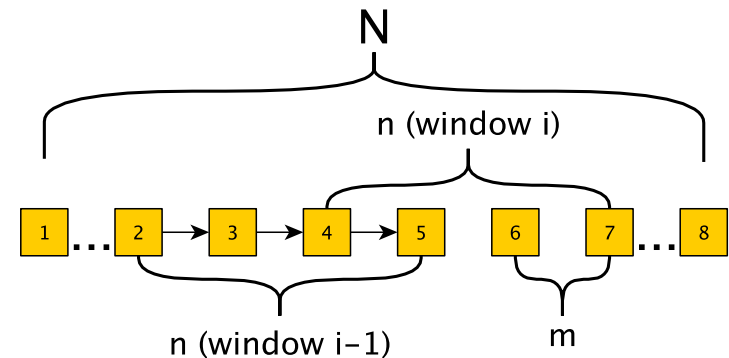
Formulation
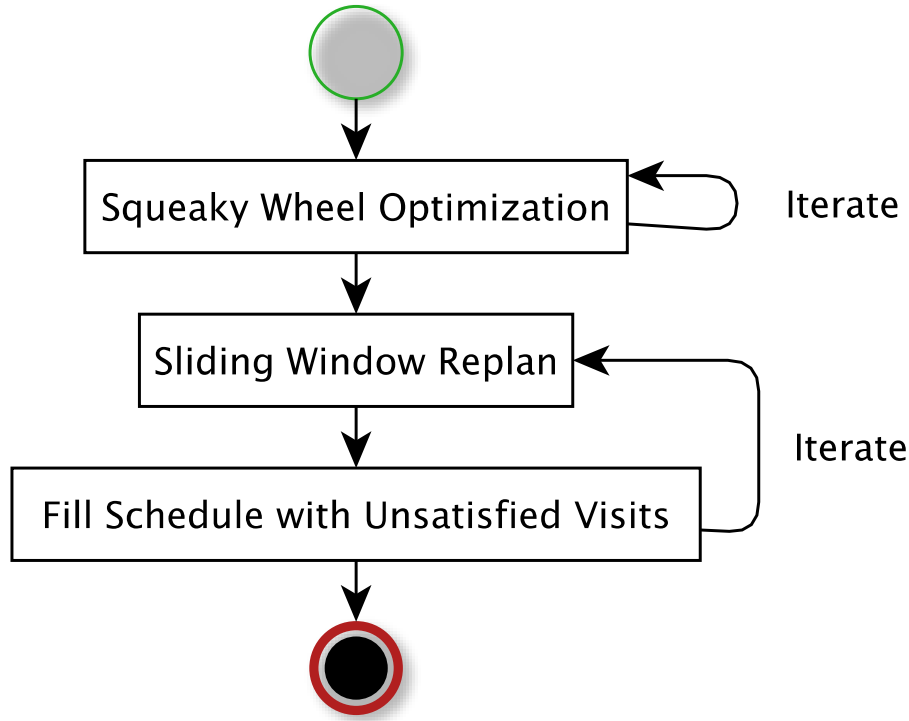
- Seed the initial schedule using SWO until convergence
- Repeat sliding window replanning with a fill phase until the schedule score doesn't increase.
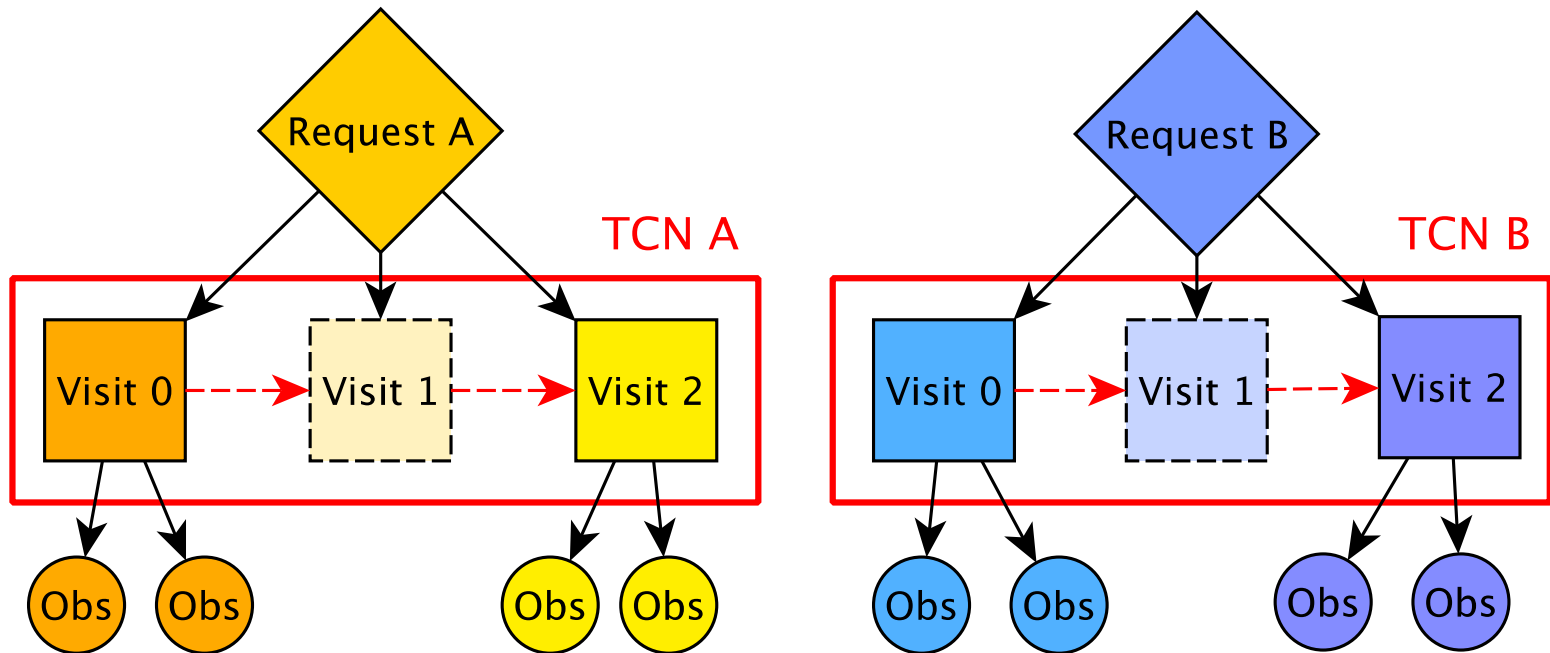
Requests, TCNs, and Visits

- Requests – User defined science targets with geometric constraints
- Visits – an atomic scheduling unit of work
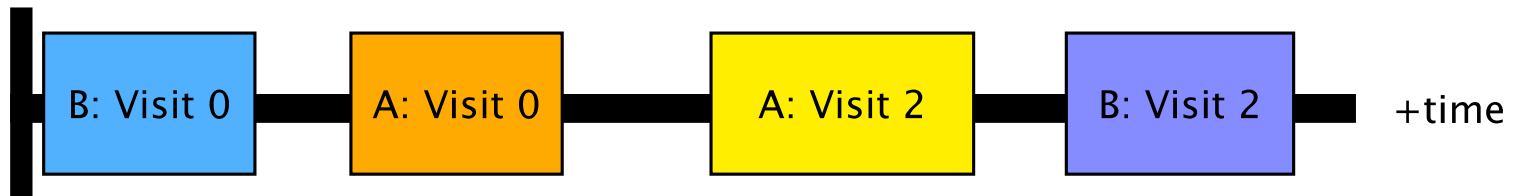- Observations – Individual frames that satisfy a visit
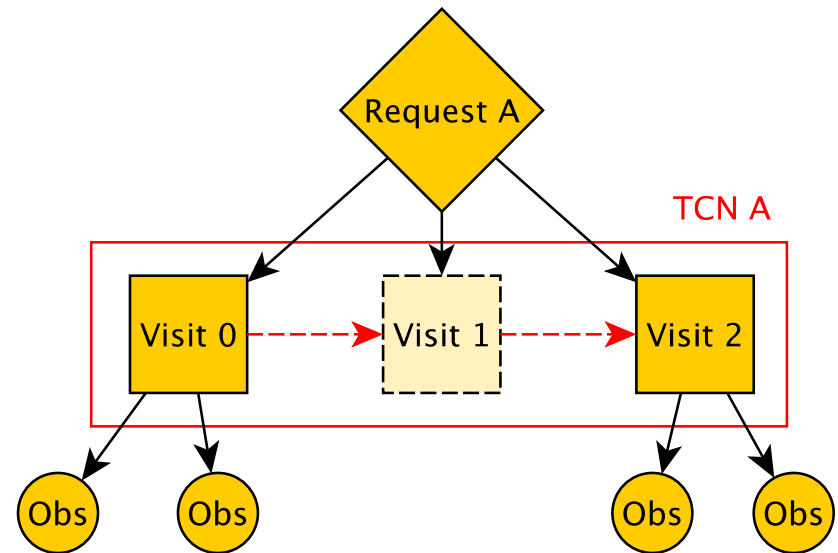
Dechter, Meiri, and Pearl 1991
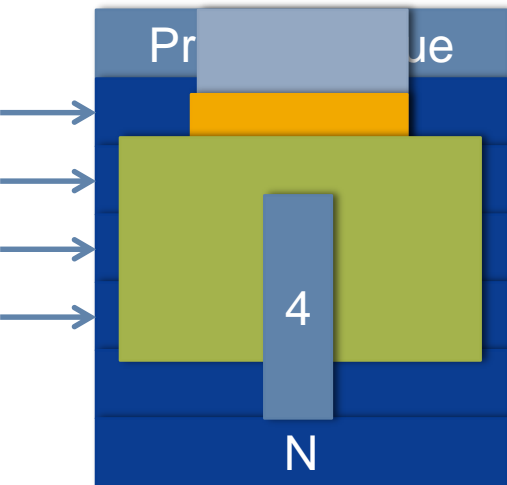
jpl.nasa.gov

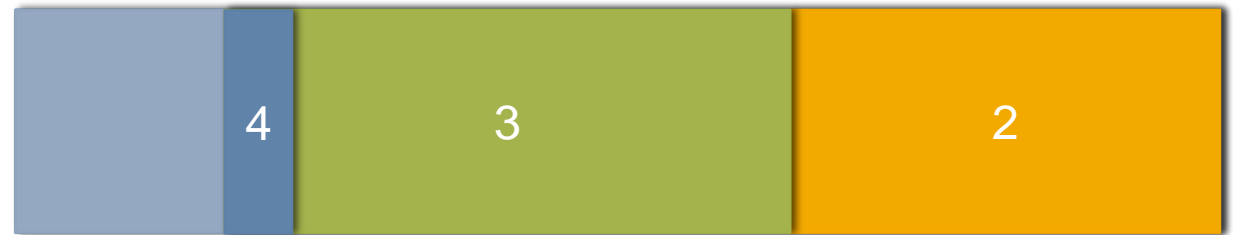# Temporal Constraint Networks

Possible Schedule (one of many)

+time

Squeaky Wheel Optimization

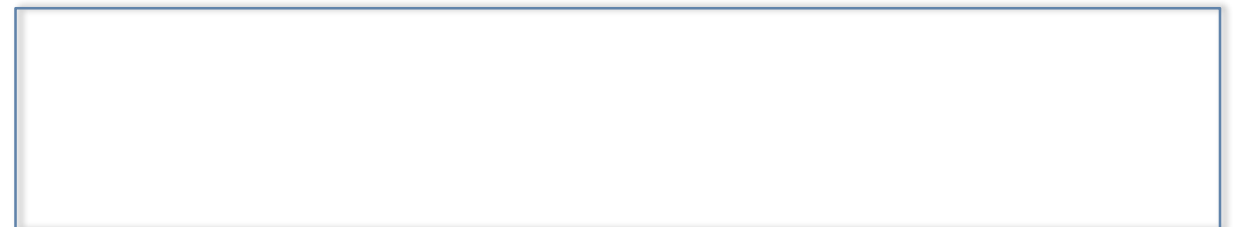Joslin and Clements 1999

2. Find all opportunities

Priority Queue

4

N

Opportunities

4 3 2

Timeline

1. Choose the highest scheduler priority request in the queue

3. Attempt to schedule in the earliest interval

4. If scheduling fails, bump scheduler priority

jpl.nasa.gov

Value-based Scoring

Priority Value Score

- Schedule is deemed more valuable if a single request with a higher priority is scheduled

$$p_{\min} = \min\left(p_i, (p_i, r_i) \in R : r_i \text{ satisified}\right)$$
$$f_{swo} = |(p_i, r_i) \in R : r_i \text{ satisified} \wedge p_i = p_{\min}|$$

Satisfaction Value Score

- Schedule score a function of % visits detailed scaled by their priority

$$f_{sat} = \sum_{i=1}^{r} w_i \frac{|v_j \in V_{TCN,i} : v_j \text{ is detailed}|}{|V_{TCN,i}|}$$

$$w_i = \begin{cases} \frac{1}{p_i}, & 1 \leq p_i \\ 2 - p_i, & p_i < 1 \end{cases}$$

Cost-based Scoring

Time Cost Score

- Penalizes idle time

$$t_{\text{cost}} = \sum (t_{start,i+1} - t_{end,i})$$

$$f_{time} = \begin{cases} \frac{1}{t_{\text{cost}}}, & 1 \leq t_{\text{cost}} \\ 2 - t_{\text{cost}}, & t_{\text{cost}} < 1| \end{cases}$$
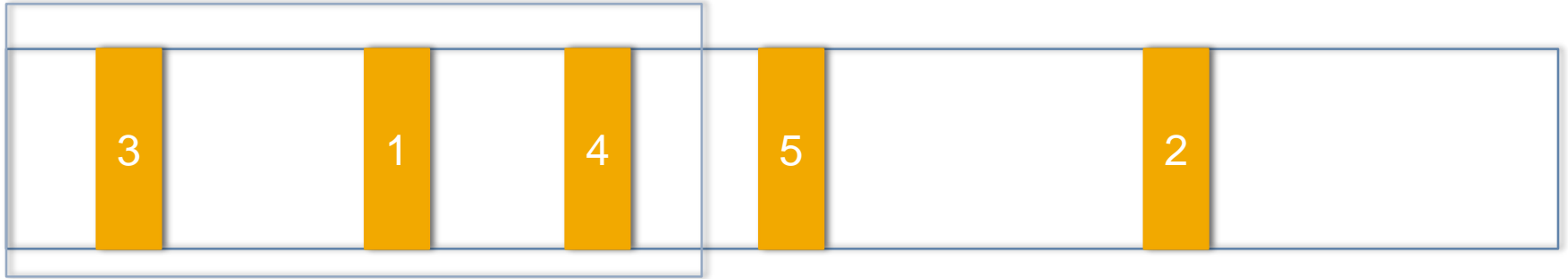
Slew Cost Score

- Penalizes larger slew angles

$$\Phi = \sum |\phi_{i,i+1}|$$

$$f_{slew} = \begin{cases} \frac{1}{\Phi}, & 1 \leq \Phi \\ 2 - \Phi, & \Phi < 1 \end{cases}$$

- The scheduler utilizes Time Cost Score when there are multiple visits from a single request within the current scheduling window
- For all other cases, Slew Cost Scoring is used

Sliding Window Replanning

- The optimal path is not constructed with sliding window replanning
- Local scope of scheduling prevents optimizations outside of the window

Insertion Heuristic

| 1 | 4 | 3 | 5 | 2 |

1 1   2   2   3

| | 1 | | 4 | | 3 | |
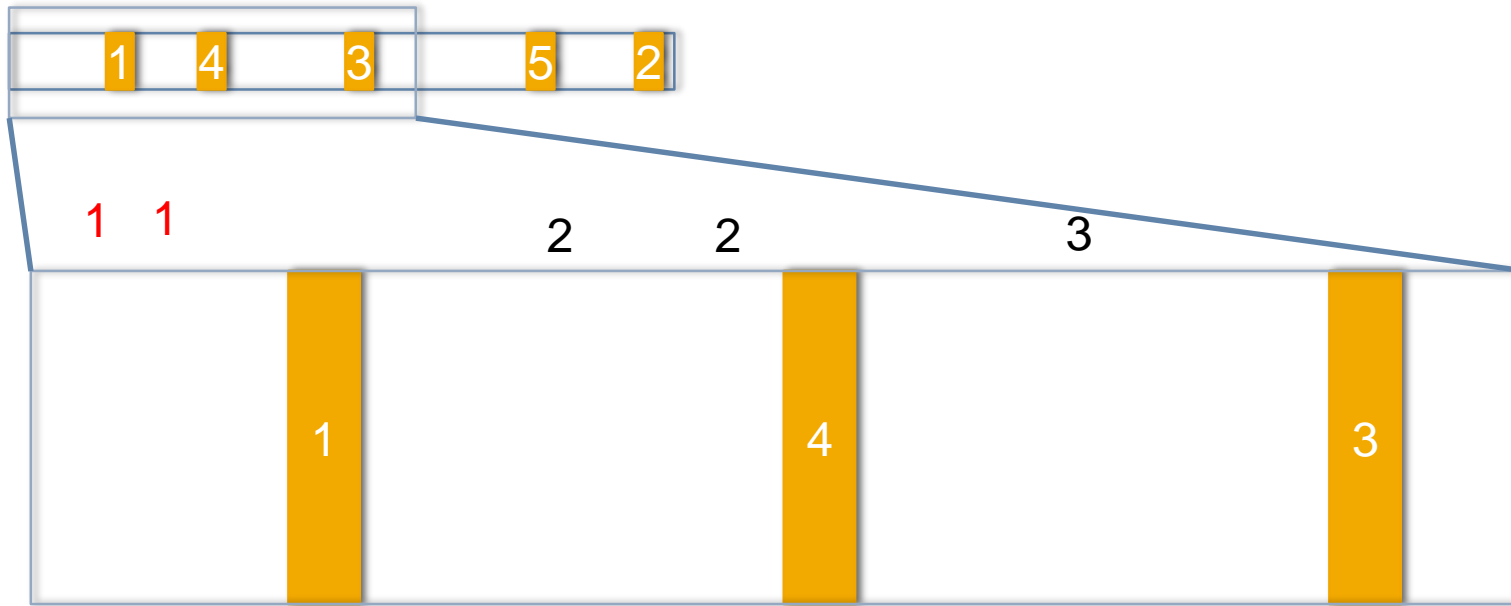
Position	Cost

Note: Scoring complications occur when a sliding window contains 2 visits of the same request

Toy Problem - Description

- 29 requests in a ring
- Carefully constructed priorities in order to force edge crossings
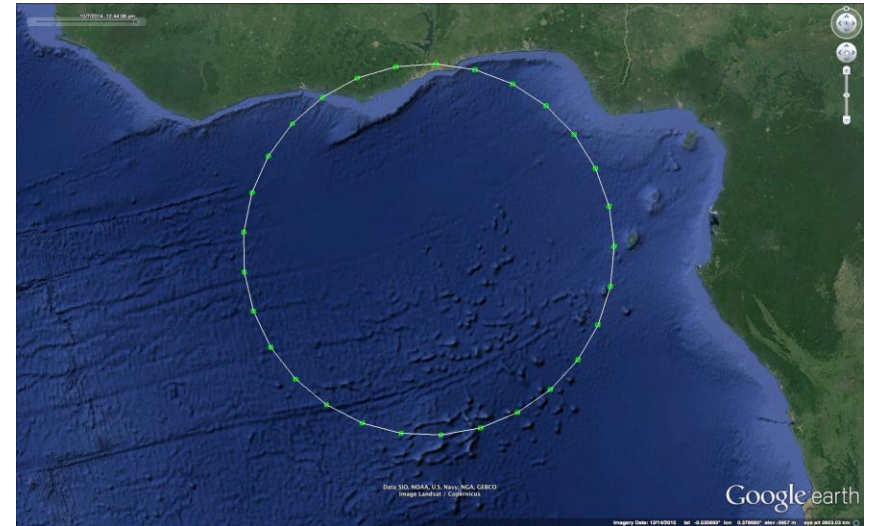- Use sliding window scheduler to fix edge crossings
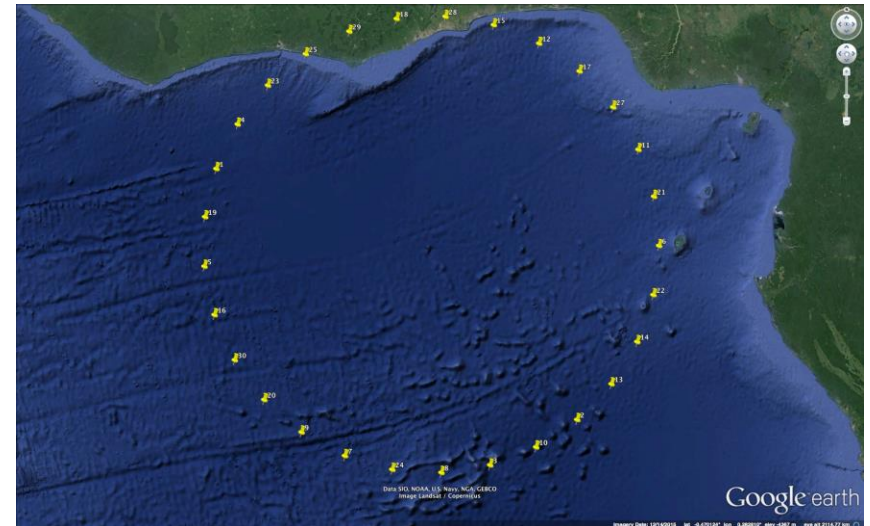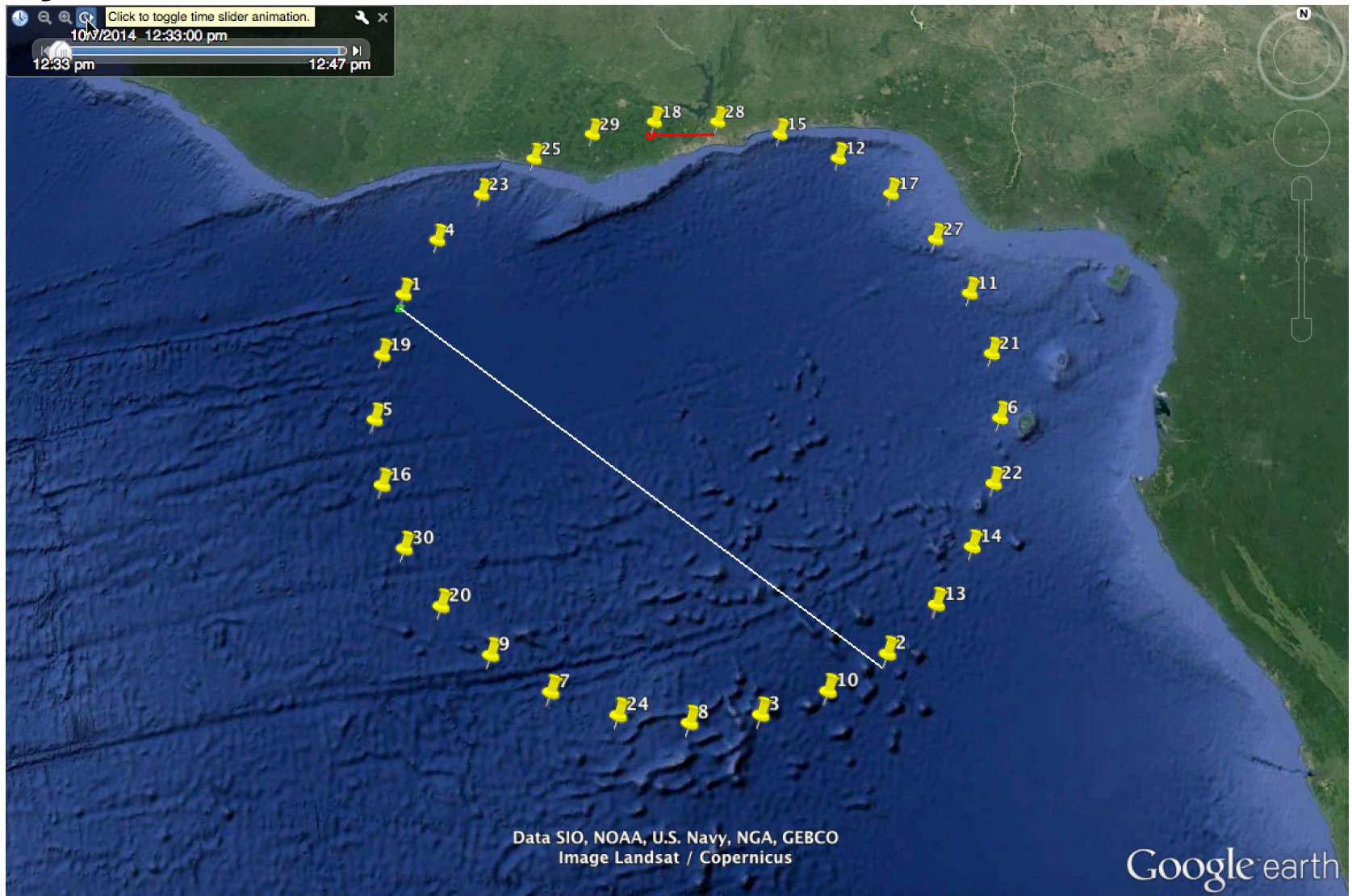
Toy Problem - Results

SWO Only

SWO-TSP Hybrid

jpl.nasa.gov

Toy Problem - Results

jpl.nasa.gov
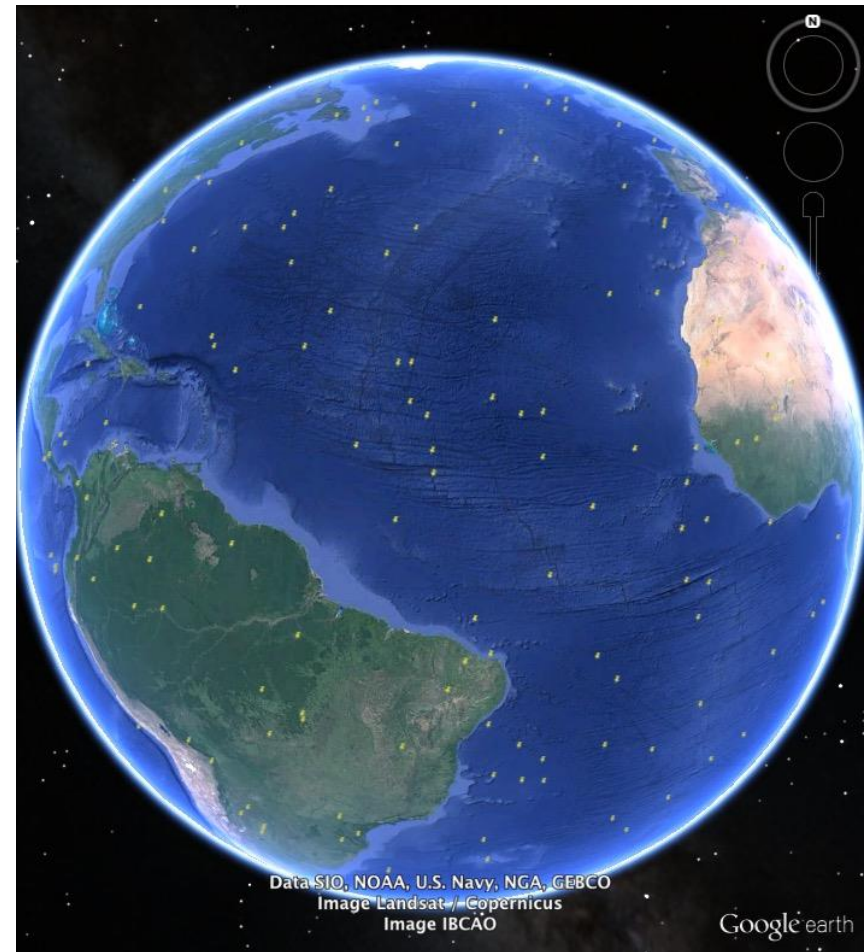
Random Points - Description

- 1000 random (uniform) point requests
- Random priorities
- Random (uniform) distribution of geometric constraints per request
- 3 agility cases (low, medium, high)

jpl.nasa.gov

Random Points - Results

- Lower agility cases see a larger improvement in request satisfaction over multiple iterations

- Algorithmic runtime costs versus the baseline schedule score
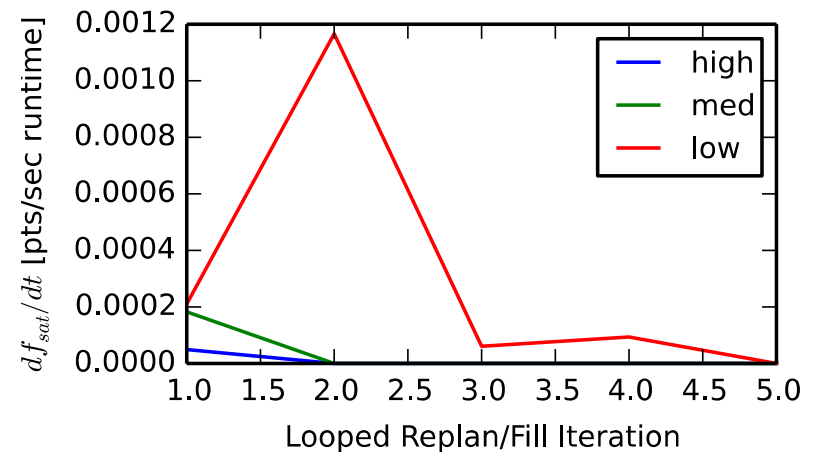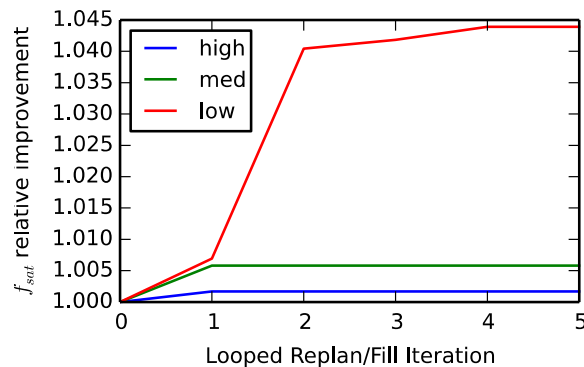
Discussion

1. TSP Replanner is Sensitive to Input Order

jpl.nasa.gov
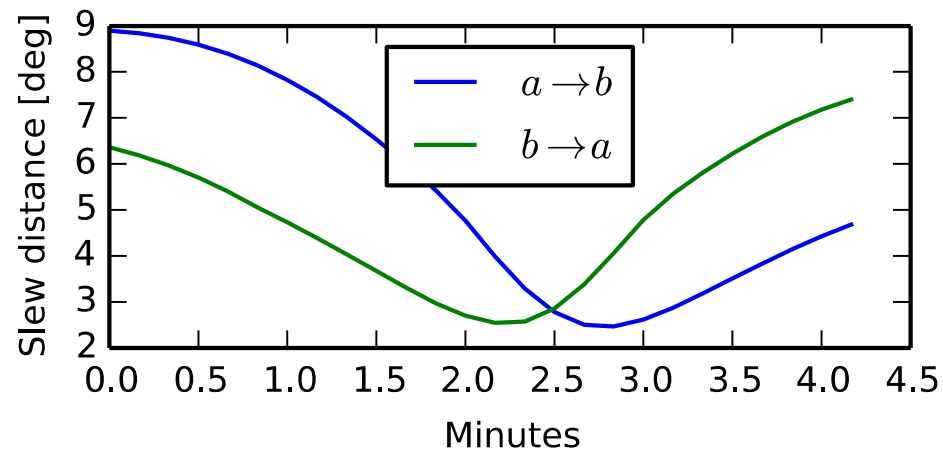
Discussion

2. Replanning may fail to maintain f_{sat}

Discussion

3. Complexity Control: Scratchpads

jpl.nasa.gov
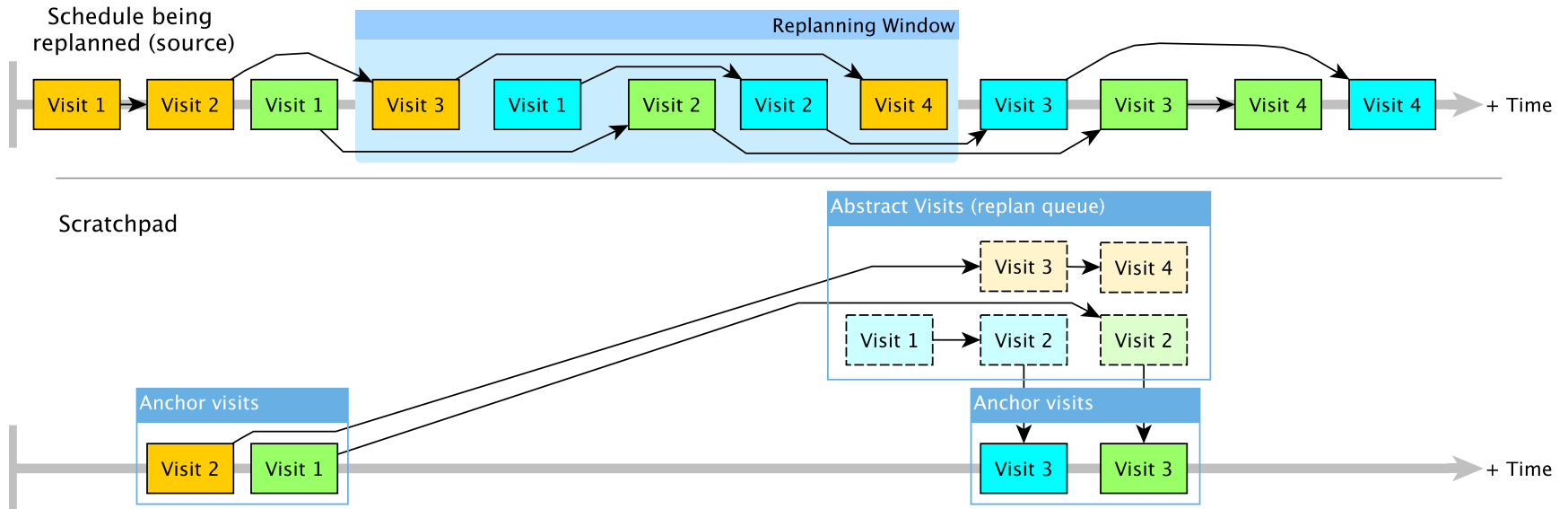
Future Work

- Initial Schedule Seeding
- Replace the insertion heuristic
- Maintain the sliding window and incrementally improve
- Different score functions

jpl.nasa.gov

Conclusion
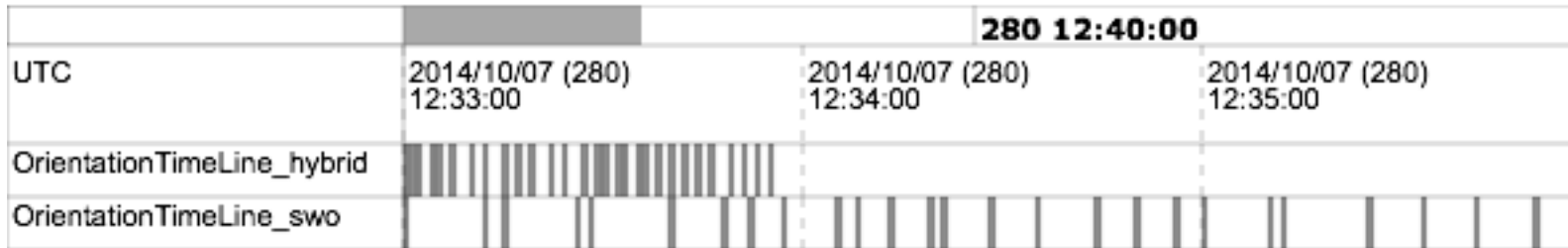
- SWO is poor at optimizing the path to satisfy requests when there is no feedback from "squeaky" requests
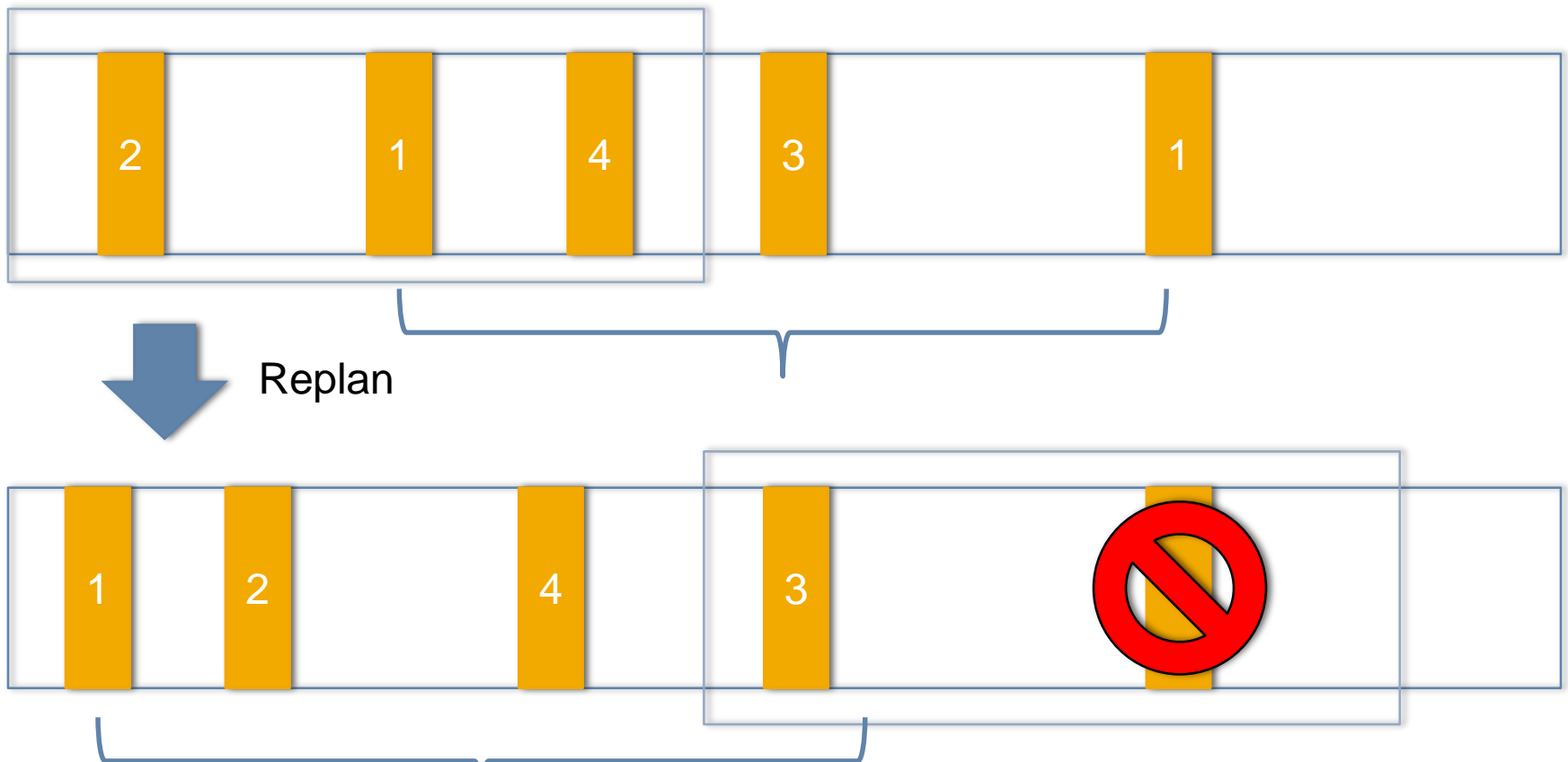
- Sliding window replanning ignores priority to improve the current seeded schedule.

- Gaps form for large or constrained requests to be satisfied during the fill phase.

- Schedule score doesn't improve much for agile systems as slew duration is no longer the constraining resource.

Acknowledgements

- The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.
- The authors would like to thank the reviewers for their constructive feedback.
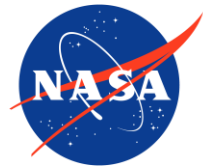
jpl.nasa.gov

References

Aldinger, J.; Lohr, J.; Winker, S.; and Willich, G. 2013. Automated planning for earth observation spacecraft under attitude dynamical constants. In *Jahrbuch der Deutschen Gesellschaft für Luft- und Raumfahrt.*

Allahverdi, A.; Ng, C.; Cheng, T. E.; and Kovalyov, M. Y. 2008. A survey of scheduling problems with setup times or costs. *European Journal of Operational Research* 187(3):985–1032.

Analytical Graphics, Inc. 2015. Algorithm definitions. Web. Accessed: 2017-03-17.

Chien, S.; Rabideau, G.; Knight, R.; Sherwood, R.; Engelhardt, B.; Mutz, D.; Estlin, T.; Smith, B.; Fisher, F.; Barrett, T.; et al. 2000. Aspen–automated planning and scheduling for space mission operations. In *International Conference on Space Operations (SpaceOps 2000)*, 1–10.

Cormen, T. H.; Leiserson, C. E.; Rivest, R. L.; and Stein, C. 2009. *Introduction to Algorithms, 3rd Edition:*. MIT Press.

Croes, G. 1958. A method for solving traveling salesman problems. *Operations Research* 6:791–812.

Dechter, R.; Meiri, I.; and Pearl, J. 1991. Temporal constraint networks. *Artificial intelligence* 49(1-3):61–95.

Frank, J.; Jonsson, A.; Morris, R.; and Smith, D. 2001. Planning and scheduling for fleets of earth observing satellites. In *In Proceedings of Sixth International Symposium on Artificial Intelligence, Robotics, Automation and Space.*

Garey, M., and Johnson, D. 1979. *Computers and Intractability: A Guide to the Theory of NP-completeness.* Freeman.

Globus, A.; Crawford, J.; Lohn, J.; and Pryor, A. 2004. A comparison of techniques for scheduling earth observing satellites. In *Proceedings of the 16th conference on Innovative Applications of Artificial Intelligence.* AAAI Press.

Goldengorin, B., and Jäger, G. 2005. How to make a greedy heuristic for the asymmetric traveling salesman problem competitive. Technical report, University Groningen.

Hall, N., and Magazine, M. 1994. Maximizing the value of a space mission. *European journal of operational research* 78:224–241.

Ichoua, S.; Gendreau, M.; and Potvin, J.-Y. 2003. Vehicle dispatching with time-dependent travel times. *European Journal of Operational Research* 144:379–396.

Joslin, D. E., and Clements, D. P. 1999. Squeaky wheel optimization. *Journal of Artificial Intelligence Research* 10:353–373.

Karger, D.; Motwani, R.; and Ramkumar, G. D. 1997. On approximating the longest path in a graph. *Algorithmica* 18(1):82–98.

Knight, R., and Chien, S. 2006. Producing large observation campaigns using compressed problem representations. In *Proceedings of the 8th International Workshop of Planning and Scheduling for Space (IWPSS-2006).*

Knight, R.; Donnellan, A.; and Green, J. J. 2013. Mission design evaluation using automated planning for high resolution imaging of dynamic surface processes from the iss. In *Proceedings of the 8th International Workshop of Planning and Scheduling for Space (IWPSS-2013).* Jet Propulsion Laboratory, National Aeronautics and Space Administration.

Lemaître, M.; Verfaillie, G.; Jouhaud, F.; Lachiver, J.-M.; and Bataille, N. 2002. Selecting and scheduling observations of agile satellites. *Aerospace Science and Technology* 6:367–381.

Pinedo, M. L. 2012. *Scheduling: Theory, Algorithms, and Systems.* Springer Science+Business Media.

Pralet, C., and Verfaillie, G. 2014. Time-dependent simple temporal networks: Properties and algorithms. In *Proceedings of the Twenty-Fourth International Conference on Automated Planning and Scheduling.*

Rabideau, G.; Chien, S.; McClaren, D.; Knight, R.; Anwar, S.; Mehall, G.; and Christensen, P. 2010. A tool for scheduling themis observations. In *In Proceedings of Sixth International Symposium on Artificial Intelligence, Robotics, Automation and Space.*

Reinelt, G. 1994. *The traveling salesman: computational solutions for TSP applications.* Springer-Verlag.

Rosenkrantz, D. J.; Stearns, R. E.; and Lewis, II, P. M. 1977. An analysis of several heuristics for the traveling salesman problem. *SIAM journal on computing* 6(3):563–581.